

In the Claims:

1. (original) A computer-implemented method for sharing one or more high-level language data structures between an assembly language program and a high-level language program, each data structure including one or more elements, comprising the steps of:

determining storage requirements from high-level language definitions of the one or more data structures in an assembly language source program and removing the definitions from the assembly source program;

determining memory addresses of the one or more data structures from memory allocation directives in the assembly language source program and removing the allocation directives from the assembly source program; and

replacing references to elements of the data structures in the assembly source program with memory addresses.

2. (original) The method of claim 1, further comprising:

assembling the assembly source program into a first object code segment;

compiling the high-level language program into a second object code segment; and

linking the first and second object code segments.

3. (original) The method of claim 1, wherein each allocation directive includes a reference to a data structure definition, a variable name, and an address.

4. (original) The method of claim 3, wherein variable name includes an array size.

5. (original) The method of claim 1, wherein a reference to an element of a data structure in the assembly source program includes a hierarchical specification of the element.

6. (original) The method of claim 5, wherein a reference to an element of a data structure in the assembly source program includes one of a request for an address and a request for an offset address of the element.

BEST AVAILABLE COPY

7. (original) The method of claim 5, wherein a reference to an element of a data structure in the assembly source program includes one of a request for an address of the element, a request for an offset address of the element, and a request for a size of the element.
8. (original) The method of claim 7, further comprising replacing a reference to an element of a data structure that includes a request for a size of an element with the size of the element.
9. (original) The method of claim 8, further comprising:
 - assembling the assembly source program into a first object code segment;
 - compiling the high-level language program into a second object code segment; and
 - linking the first and second object code segments.
10. (original) The method of claim 8, wherein each allocation directive includes a reference to a data structure definition, a variable name, and an address.
11. (original) The method of claim 10, wherein variable name includes an array size.
12. (original) The method of claim 5, further comprising:
 - assembling the assembly source program into a first object code segment;
 - compiling the high-level language program into a second object code segment; and
 - linking the first and second object code segments.
13. (original) The method of claim 5, wherein each allocation directive includes a reference to a data structure definition, a variable name, and an address.
14. (original) An apparatus for sharing one or more high-level language data structures between an assembly language program and a high-level language program, each data structure including one or more elements, comprising:

BEST AVAILABLE COPY

means for determining storage requirements from high-level language definitions of the one or more data structures in an assembly language source program and removing the definitions from the assembly source program;

means for determining memory addresses of the one or more data structures from memory allocation directives in the assembly language source program and removing the allocation directives from the assembly source program; and

means for replacing references to elements of the data structures in the assembly source program with memory addresses.

15. (previously presented) The method of claim 13, wherein variable name includes an array size.

BEST AVAILABLE COPY